# WEST Search History

| Hide Items | Restore | Clear | Cancel |

DATE: Tuesday, March 29, 2005

| Hide? | Set Name | Query | Hit Count |
|---|---|---|---|
| | | *DB=PGPB, USPT, USOC, EPAB, JPAB, DWPI, TDBD; PLUR=YES; OP=ADJ* | |
| ☐ | L11 | 717/$.ccls. and 110 | 61 |
| ☐ | L10 | (compil$6 or translat$6) with (code$2 or program$2 or language$2) and (source with compil$6 with rule$2) | 186 |
| ☐ | L9 | (regenerat$6 with compil$6) | 153 |
| | | *DB=PGPB, USOC, EPAB, JPAB, DWPI, TDBD; PLUR=YES; OP=ADJ* | |
| ☐ | L8 | 16 and L7 | 5 |
| ☐ | L7 | (compil$6 or translat$6) with (code$2 or program$2 or language$2) and (search$3 with path$2) | 454 |
| ☐ | L6 | 717/140-145.ccls. | 342 |
| | | *DB=USPT; PLUR=YES; OP=ADJ* | |
| ☐ | L5 | L4 and 11 | 22 |
| ☐ | L4 | (compil$6 or translat$6) with (code$2 or program$2 or language$2) and (search$3 with path$2) | 680 |
| ☐ | L3 | L2 and 11 | 15 |
| ☐ | L2 | (compile$6 or translat$6) with source and (search$3 with path$2) | 235 |
| ☐ | L1 | 717/140-145.ccls. | 730 |

END OF SEARCH HISTORY

Subscribe (Full Service)   Register (Limited Service, Free)   Login

**Search:**   ⊙ The ACM Digital Library   ○ The Guide

| +abstract:compilation +abstract:source +abstract:rule | SEARCH |

**THE ACM DIGITAL LIBRARY**

☞ Feedback  Report a problem  Satisfaction survey

Terms used **compilation source rule**                    Found **13** of **151,219**

Sort results by   `relevance ▼`          🔖 Save results to a Binder          Try an **Advanced Search**

Display results   `expanded form ▼`      ❓ Search Tips                         Try this search in The ACM Guide

☐ Open results in a new window

Results 1 - 13 of 13

Relevance scale ☐ ☐ ▬ ▬ ▬

### 1   The GNAT compilation model
Robert Dewar
November 1994 **Proceedings of the conference on TRI-Ada '94**

Full text available: 📄 pdf(1.55 MB)      Additional Information: full citation, abstract, references, citings, index terms

One of the novel features of GNAT is its unusual approach to the compilation process and the handling of the Ada library. The words novel and unusual only apply from a traditional Ada compilation perspective. By contrast, a typical C or C++ programmer would find many aspects of the model quite familiar. In GNAT, sources are independently compiled to produce a set of objects, and the set of object files thus produced is submitted to the binder/linker to generate the resulting executable. Thi ...

### 2   Verification of the application of coding rules for Ada to enhance portability of real-time applications
P. Obermayer, J. Schröer, R. A. Peek, N. Collienne, A. Klimek, R. Landwehr
July 1994   **Proceedings of the eleventh annual Washington Ada symposium & summer ACM SIGAda meeting on Ada**

Full text available: 📄 pdf(366.89 KB)    Additional Information: full citation, abstract, references, index terms

In order to improve the portability of code for real time applications written in Ada, a set of general coding rules has been compiled. In a second step, Ada code constructs offending these rules were identified. In order to support the verification of the application of coding rules a static analysis tool was developed, which tests Ada source code for these constructs and which indicates the offending constructs. Since a syntactical analysis is insufficient to perform all necessary tests, ...

### 3   Adding rule-based techniques to procedural languages
Keith R. Milliken, Allan J. Finkel, David A. Klein, Norman B. Waite
June 1988 **Proceedings of the first international conference on Industrial and engineering applications of artificial intelligence and expert systems - Volume 1**

Full text available: 📄 pdf(906.00 KB)      Additional Information: full citation, abstract, references, citings, index terms

The world of software for applications in industry is dominated by compiled, algebraic (Algol derivative) programming languages such as C, Pascal, and PL/I. One approach to popularizing expert systems techniques in the industrial environment is to integrate software techniques appropriate for expert systems development into algebraic

programming languages. While programmers in industry seek more powerful software tools, generally they do not want to give up existing capabilities in ...

**4** <u>A type system for Java bytecode subroutines</u>

Raymie Stata, Martin Abadi

January 1999 **ACM Transactions on Programming Languages and Systems (TOPLAS)**,
Volume 21 Issue 1

Full text available: pdf(519.84 KB)    Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Java is typically compiled into an intermediate language, JVML, that is interpreted by the Java Virtual Machine. Because mobile JVML code is not always trusted, a bytecode verifier enforces static constraints that prevent various dynamic errors. Given the importance of the bytecode verifier for security, its current descriptions are inadequate. This article proposes using typing rules to describe the bytecode verifier because they are more precise than prose, clearer than code, and easier ...

**Keywords**: Java, bytecode verification

**5** <u>PROFGEN: a procedure for generating machine independent high-level language profilers</u>

Paul A. Fishwick, Stefan Feyock

April 1984 **ACM SIGMETRICS Performance Evaluation Review**, Volume 12 Issue 2

Full text available: pdf(319.28 KB)    Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>

Many of the tools used in software metrics for evaluating the execution characteristics of a program are predicated on specific counting rules for operators and operands [1, 2]. The analyst may use these counting techniques to determine such program attributes as estimation of object code size prior to actual compilation and the relative efficiencies of various language compilers. Operator/operand measures provide useful results for certain analyses, but a deficiency exists in that the data deri ...

**6** <u>Using meta-level compilation to check FLASH protocol code</u>

Andy Chou, Benjamin Chelf, Dawson Engler, Mark Heinrich

November 2000 **Proceedings of the ninth international conference on Architectural support for programming languages and operating systems**, Volume 34 ,
28 Issue 5 , 5

Full text available: pdf(297.02 KB)    Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Building systems such as OS kernels and embedded software is difficult. An important source of this difficulty is the numerous rules they must obey: interrupts cannot be disabled for ~too long," global variables must be protected by locks, user pointers passed to OS code must be checked for safety before use, etc. A single violation can crash the system, yet typically these invariants are unchecked, existing only on paper or in the implementor's mind.This paper is a case study in how system impl ...

**7** <u>Using meta-level compilation to check FLASH protocol code</u>

Andy Chou, Benjamin Chelf, Dawson Engler, Mark Heinrich

November 2000 **ACM SIGPLAN Notices**, Volume 35 Issue 11

Full text available: pdf(1.39 MB)    Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>index terms</u>

Building systems such as OS kernels and embedded software is difficult. An important source of this difficulty is the numerous rules they must obey: interrupts cannot be disabled for "too long," global variables must be protected by locks, user pointers passed to OS code must be checked for safety before use, etc. A single violation can crash the system, yet

, typically these invariants are unchecked, existing only on paper or in the implementor's mind.This paper is a case study in how system impl ...

**8** <u>PCDEC, an interactive decision table system for personal computers (abstract only)</u>
Begonia Bi-Hsia Tai, Leonard H. Weiner
February 1987 **Proceedings of the 15th annual conference on Computer Science**

Full text available: <u>pdf(104.00 KB)</u>     Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

To date, most of the effort in computer creation of decision tables (DTs) has been directed toward automatic conversion of tables into program source code, which is then compiled and executed in batch mode. This preprocessor approach to processing decision tables is usually implemented on a mainframe computer and needs programmers to prepare the input for the proprocessor. As a result, using the computer to create decision tables is difficult for those without a good programming background. ...

**9** <u>Enriching the lambda calculus with contexts: toward a theory of incremental program construction</u>
Shinn-Der Lee, Daniel P. Friedman
June 1996 **ACM SIGPLAN Notices , Proceedings of the first ACM SIGPLAN international conference on Functional programming**, Volume 31 Issue 6

Full text available: <u>pdf(1.32 MB)</u>     Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

A context in the &lambda;-calculus is a term with some holes. Hole filling differs from &beta;-substitution in that name capture is intended. This seemingly simple feature transcends static scope and lies at the heart of modular and object-oriented programming. Still, the name capture feature of hole filling is at odds with hygienic &beta;-substitution. In this paper we conservatively extend the &lambda;-calculus to incorporate the notion of contexts without jeopardizing the & ...

**10** <u>Retargetable high-level alias analysis</u>
Deborah S. Coutant
January 1986 **Proceedings of the 13th ACM SIGACT-SIGPLAN symposium on Principles of programming languages**

Full text available: <u>pdf(1.53 MB)</u>     Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>

All optimizing compilers must deal with the problem of aliases arising due to the presence of multiple names that reference the same memory areas. Presented in this paper is a staged, high-level alias analysis methodology that provides detailed alias information to a global optimizer implemented at any level in the compilation process. The framework provides easy portability of optimizing compilers to new architectures, as well as the easy addition, of new compilers to an already existing family ...

**11** <u>Design of LMT: a prolog-based machine translation system</u>
Michael C. McCord
March 1989 **Computational Linguistics**, Volume 15 Issue 1

Full text available: <u>pdf(2.36 MB)</u>     Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>
<u>Publisher Site</u>

LMT (logic-based machine translation) is an experimental English-to-German MT system, being developed in the framework of logic programming. The English analysis uses a logic grammar formalism, Modular Logic Grammar, which allows logic grammars to be more compact, and which has a modular treatment of syntax, lexicon, and semantics. The English grammar is written independently of the task of translation. LMT uses a syntax transfer method for translation, although the English syntactic analysis tr ...

**12** Oracle-based checking of untrusted software

George C. Necula, S. P. Rahul

January 2001 **ACM SIGPLAN Notices , Proceedings of the 28th ACM SIGPLAN-SIGACT symposium on Principles of programming languages**, Volume 36 Issue 3

Full text available: pdf(823.41 KB)    Additional Information: full citation, abstract, references, citings, index terms

We present a variant of Proof-Carrying Code (PCC) in which the trusted inference rules are represented as a higherorder logic program, the proof checker is replaced by a nondeterministic higher-order logic interpreter and the proof by an oracle implemented as a stream of bits that resolve the nondeterministic interpretation choices. In this setting, Proof-Carrying Code allows the receiver of the code the luxury of using nondeterminism in constructing a simple yet powerful checking procedure.This ...

**13** Writing a compilers compiler in APL

Enrique Alfonseca

July 1998  **ACM SIGAPL APL Quote Quad , Proceedings of the APL98 conference on Array processing language**, Volume 29 Issue 3

Full text available: pdf(393.34 KB)    Additional Information: full citation, abstract, references, index terms

This paper describes the special capabilities of APL2 for the construction of compilers and translator writing systems (TWS). To that purpose, a TWS has been written in APL2, which takes as input a complete description of a language including an attribute grammar, in extended Backus normal form, and the semantic function associated to each rule of the grammar, written in APL2. The output of the TWS is a compiler, generated directly inside the workspace and ready to compile source code written in ...

Results 1 - 13 of 13

Useful downloads: Adobe Acrobat    QuickTime    Windows Media Player    Real Player

# PORTAL

**US Patent & Trademark Office.**

**Search:**   ◉ The ACM Digital Library   ○ The Guide

+abstract:compilation +abstract:search +abstract:rule

**THE ACM DIGITAL LIBRARY**

📷 Feedback  Report a problem  Satisfaction survey

Terms used **compilation search rule**                                    Found **6** of **151,219**

Sort results by   relevance ▼

Display results   expanded form ▼

📖 Save results to a Binder

❓ Search Tips

☐ Open results in a new window

Try an **Advanced Search**

Try this search in **The ACM Guide**

Results 1 - 6 of 6

Relevance scale ☐ ▭ ▬ ▬ ■

**1** A comparison of receiver-initiated and sender-initiated adaptive load sharing (extended abstract)
Derek L. Eager, Edward D. Lazowska, John Zahorjan
August 1985 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1985 ACM SIGMETRICS conference on Measurement and modeling of computer systems**, Volume 13 Issue 2
Full text available: 📄 pdf(351.78 KB)     Additional Information: full citation, abstract, references, citings, index terms

One goal of locally distributed systems is to facilitate resource sharing. Most current locally distributed systems, however, share primarily data, data storage devices, and output devices; there is little sharing of computational resources. Load sharing is the process of sharing computational resources by transparently distributing the system workload. System performance can be improved by transferring work from nodes that are heavily loaded to nodes that ...

**2** A pattern compiler
Richard D. Jenks
August 1976 **Proceedings of the third ACM symposium on Symbolic and algebraic computation**
Full text available: 📄 pdf(634.58 KB)     Additional Information: full citation, abstract, references, citings, index terms

A pattern compiler for the SCRATCHPAD system provides an efficient implementation of sets of user-defined pattern-replacement rules for symbolic mathematical computation such as tables of integrals or summation identities. Rules are compiled together, with common search paths merged and factored out and with the resulting code optimized for efficient recognition over all patterns. Matching principally involves structural comparison of expression trees and evaluation of predicates. Pattern r ...

**3** Posters: By the people now, for the people later: using transitory metadata to anchor a digital archive
Anne Washington
July 2002 **Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries**
Full text available: 📄 pdf(172.83 KB)     Additional Information: full citation, abstract, index terms

The Congressional Research Service (CRS) serves Congress by providing timely, objective and non_partisan research, analysis and information services. The Legislative Information Office within CRS fulfills that mandate by maintaining a digital library of legislative documents known as the Legislative Information System.An ongoing challenge is designing

these full text and structured databases for both promptness and permanence. This is accomplished by metadata and interface design. This foundatio ...

**Keywords:** digital archive, embedded links, legislation, legislative history, permanent access

**4** Assumption analysis in compiled logic-based decision support systems (LDSSs) (abstract only)
Michael C. Chen, Lawrence J. Henschen
March 1985 **Proceedings of the 1985 ACM thirteenth annual conference on Computer Science**
Additional Information: full citation, abstract, citings, index terms

We describe a LDSS which extends and improves on prior proposals1. It uses a connection graph to compile programs for handling ordinary queries2, thus eliminating the need for deductive search at query time. It provides for assumption analyses (like "what-if" questions) by allowing changes to the rules (clauses) of the system and tracing the effects of such changes, thereby avoiding the need to recompile the whole system3

**5** Papers: A compilation-chart method for linear categorial deduction
Mark Hepple
August 1996 **Proceedings of the 16th conference on Computational linguistics - Volume 1**
Full text available: pdf(589.50 KB)     Additional Information: full citation, abstract, references

Recent work in categorial grammar has seen proposals for a wide range of systems, differing in their 'resource sensitivity' and hence, implicitly, their underlying notion of 'linguistic structure'. A common framework for parsing such systems is emerging, whereby some method of linear logic theorem proving is used in combination with a system of labelling that ensures that only deductions appropriate to the relevant categorial formalism are allowed. This paper presents a deduction method for impl ...

**6** Oracle-based checking of untrusted software
George C. Necula, S. P. Rahul
January 2001 **ACM SIGPLAN Notices , Proceedings of the 28th ACM SIGPLAN-SIGACT symposium on Principles of programming languages,** Volume 36 Issue 3
Full text available: pdf(823.41 KB)     Additional Information: full citation, abstract, references, citings, index terms

We present a variant of Proof-Carrying Code (PCC) in which the trusted inference rules are represented as a higherorder logic program, the proof checker is replaced by a nondeterministic higher-order logic interpreter and the proof by an oracle implemented as a stream of bits that resolve the nondeterministic interpretation choices. In this setting, Proof-Carrying Code allows the receiver of the code the luxury of using nondeterminism in constructing a simple yet powerful checking procedure.This ...

Results 1 - 6 of 6

**Welcome United States Patent and Trademark Office**

**Search Results**             BROWSE          SEARCH          **IEEE XPLORE GUIDE**

Results for "( compiler<in>ab ) <and> ( rule<in>ab ) <and> ( target<in>ab )"          ✉e-mail
Your search matched **13** of **1137806** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

» **Key**

| | |
|---|---|
| **IEEE JNL** | IEEE Journal or Magazine |
| **IEE JNL** | IEE Journal or Magazine |
| **IEEE CNF** | IEEE Conference Proceeding |
| **IEE CNF** | IEE Conference Proceeding |
| **IEEE STD** | IEEE Standard |

**Modify Search**

( compiler<in>ab ) <and> ( rule<in>ab ) <and> ( target<in>ab )          »

☐ Check to search only within this results set

**Display Format:**   ◉ Citation   ○ Citation & Abstract

**Select**          **Article Information**

☐    1. **Implementing temporal integrity constraints using an active DBMS**
Chomicki, J.; Toman, D.;
Knowledge and Data Engineering, IEEE Transactions on
Volume 7, Issue 4, Aug. 1995 Page(s):566 - 582

AbstractPlus | References | Full Text: PDF(1408 KB)   **IEEE JNL**

☐    2. **DISIPLE: digital signal processor programming language and environment**
Dunn, S.M.; Peters, J.E.; Finkel, B.; Neafsey, L.;
Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Proc
Transactions on
Volume 38, Issue 11, Nov. 1990 Page(s):2001 - 2003

AbstractPlus | Full Text: PDF(292 KB)   **IEEE JNL**

☐    3. **C Compiler Retargeting Based on Instruction Semantics Models**
Ceng, J.; Hohenauer, M.; Leupers, R.; Ascheid, G.; Meyr, H.; Braun, G.;
Design, Automation and Test in Europe, 2005. Proceedings
07-11 March 2005 Page(s):1150 - 1155

AbstractPlus | Full Text: PDF(192 KB)   **IEEE CNF**

☐    4. **A graph-based silicon compiler for concurrent VLSI systems**
Bergamaschi, R.; Allerton, D.J.;
CompEuro '88. 'Design: Concepts, Methods and Tools'
11-14 April 1988 Page(s):36 - 47

AbstractPlus | Full Text: PDF(660 KB)   **IEEE CNF**

☐    5. **A compiler that easily retargets high level language programs for different signal architectures**
Peters, J.E.; Dunn, S.M.;
Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conf
23-26 May 1989 Page(s):1103 - 1106 vol.2

AbstractPlus | Full Text: PDF(320 KB)   **IEEE CNF**

☐    6. **Conditional lambda-theories and the verification of static properties of programs**
Wand, M.; Wang, Z.-Y.;
Logic in Computer Science, 1990. LICS '90, Proceedings., Fifth Annual IEEE Symposiu
4-7 June 1990 Page(s):321 - 332

AbstractPlus | Full Text: PDF(588 KB)   **IEEE CNF**

☐ 7. **Synthesis of parallel Ada code from a knowledge base of rules**
Chun, R.; Lichota, R.; Perry, B.; Sabha, N.;
Parallel and Distributed Processing, 1991. Proceedings of the Third IEEE Symposium o
2-5 Dec. 1991 Page(s):600 - 607

AbstractPlus | Full Text: PDF(688 KB)  **IEEE CNF**

☐ 8. **DML-a meta-language and system for the generation of practical and efficient com denotational specifications**
Pattersson, M.; Fritzson, P.;
Computer Languages, 1992., Proceedings of the 1992 International Conference on
20-23 April 1992 Page(s):127 - 136

AbstractPlus | Full Text: PDF(760 KB)  **IEEE CNF**

☐ 9. **A retargetable technique for predicting execution time**
Harmon, M.G.; Baker, T.P.; Whalley, D.B.;
Real-Time Systems Symposium, 1992
2-4 Dec. 1992 Page(s):68 - 77

AbstractPlus | Full Text: PDF(716 KB)  **IEEE CNF**

☐ 10. **A simple tree pattern matching algorithm for code generator**
Tzer-Shyong Chen; Feipei Lai; Rung-Ji Shang;
Computer Software and Applications Conference, 1995. COMPSAC 95. Proceedings.,
Annual International
9-11 Aug. 1995 Page(s):162 - 167

AbstractPlus | Full Text: PDF(420 KB)  **IEEE CNF**

☐ 11. **Using fuzzy logic to derive an optimum embedded application specific architectu**
Nguyen, A.; Nguyen, C.;
WESCON/'95. Conference record. 'Microelectronics Communications Technology Prod
Products Mobile and Portable Power Emerging Technologies'
7-9 Nov. 1995 Page(s):618

AbstractPlus | Full Text: PDF(440 KB)  **IEEE CNF**

☐ 12. **VisPro: a visual language generation toolset**
Da-Qian Zhang; Kang Zhang;
Visual Languages, 1998. Proceedings. 1998 IEEE Symposium on
1-4 Sept. 1998 Page(s):195 - 202

AbstractPlus | Full Text: PDF(76 KB)  **IEEE CNF**

☐ 13. **A standard cell library for student projects**
Grad, J.; Stine, J.E.;
Microelectronic Systems Education, 2003. Proceedings. 2003 IEEE International Confe
1-2 June 2003 Page(s):98 - 99

AbstractPlus | Full Text: PDF(211 KB)  **IEEE CNF**

View Selected Items

Help    Contact Us    Privacy & S

**IEEE Xplore®**
RELEASE 2.0

**Welcome United States Patent and Trademark Office**

**Search Results**

BROWSE        SEARCH        IEEE XPLORE GUIDE

Results for "( recompile<in>ab ) <and> ( source<in>ab ) <and> ( rule<in>ab )"          ✉ e-mail
Your search matched **0** of **1137806** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» <u>View Session History</u>

» <u>New Search</u>

» **Key**

**Modify Search**

( recompile<in>ab ) <and> ( source<in>ab ) <and> ( rule<in>ab )          » 

| IEEE JNL | IEEE Journal or Magazine |
| IEE JNL | IEE Journal or Magazine |

☐ Check to search only within this results set

**Display Format:**    ◉ Citation    ○ Citation & Abstract

| IEEE CNF | IEEE Conference Proceeding |
| IEE CNF | IEE Conference Proceeding |

**No results were found.**

| IEEE STD | IEEE Standard |

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revisin

Help    Contact Us    Privacy & S

Indexed by
**Inspec®**

**IEEE** *Xplore* ®
RELEASE 2.0

**Welcome United States Patent and Trademark Office**

**Search Results**

BROWSE          SEARCH          IEEE XPLORE GUIDE

Results for "( recompile<in>ab ) <and> ( source<in>ab )"          ✉ e-mail
Your search matched **9** of **1137806** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

**» Key**

**Modify Search**

( recompile<in>ab ) <and> ( source<in>ab )          »

☐ Check to search only within this results set

**Display Format:**   ◉ Citation   ○ Citation & Abstract

| | |
|---|---|
| **IEEE JNL** | IEEE Journal or Magazine |
| **IEE JNL** | IEE Journal or Magazine |
| **IEEE CNF** | IEEE Conference Proceeding |
| **IEE CNF** | IEE Conference Proceeding |
| **IEEE STD** | IEEE Standard |

**Select**      **Article Information**

☐   1. **Software maintenance during system testing: enhanced methods for the System**
Parish, R.J.;
Selected Areas in Communications, IEEE Journal on
Volume 6,  Issue 8,  Oct. 1988 Page(s):1385 - 1392

AbstractPlus | Full Text: PDF(848 KB)   **IEEE JNL**

☐   2. **Welcome to the opportunities of binary translation**
Altman, E.R.; Kaeli, D.; Sheffer, Y.;
Computer
Volume 33,  Issue 3,  March 2000 Page(s):40 - 45

AbstractPlus | References | Full Text: PDF(647 KB)   **IEEE JNL**

☐   3. **PA-RISC to IA-64: transparent execution, no recompilation**
Zheng, C.; Thompson, C.;
Computer
Volume 33,  Issue 3,  March 2000 Page(s):47 - 52

AbstractPlus | Full Text: PDF(276 KB)   **IEEE JNL**

☐   4. **Monte Carlo image reconstruction of a mammographic phantom**
Ramos, M.; Ferrer, S.; Villaescusa, J.I.; Verdu, G.; Campayo, J.M.;
Engineering in Medicine and Biology Society, 2004. EMBC 2004. Conference Proceedi
International Conference of the
Volume 1,  1-5 Sept. 2004 Page(s):1467 - 1470 Vol.2

AbstractPlus | Full Text: PDF(264 KB)   **IEEE CNF**

☐   5. **A user-level process package for PVM**
Konuru, R.; Casas, J.; Prouty, R.; Otto, S.; Walpole, J.;
Scalable High-Performance Computing Conference, 1994. Proceedings of the
23-25 May 1994 Page(s):48 - 55

AbstractPlus | Full Text: PDF(788 KB)   **IEEE CNF**

☐   6. **Expert system for control purpose based on CLIPS**
Simo, J.; Martinez, M.; Morant, F.; Crespo, A.;
Electrotechnical Conference, 1994. Proceedings., 7th Mediterranean
12-14 April 1994 Page(s):254 - 257 vol.1

AbstractPlus | Full Text: PDF(272 KB)   **IEEE CNF**

☐  7. **Experience with user interface reengineering: transferring DOS panels to Window**
Csaba, L.;
Software Maintenance and Reengineering, 1997. EUROMICRO 97., First Euromicro Co
17-19 March 1997 Page(s):150 - 155
AbstractPlus | Full Text: PDF(940 KB)   **IEEE CNF**

☐  8. **Ada 95 and software reuse**
Rice, G.F.; Corcoran, S.; Leiberman, D.; Powers, R.;
Digital Avionics Systems Conference, 1997. 16th DASC., AIAA/IEEE
Volume 1,  26-30 Oct. 1997 Page(s):1.1 - 11-18 vol.1
AbstractPlus | Full Text: PDF(836 KB)   **IEEE CNF**

☐  9. **A practical method for watermarking Java programs**
Monden, A.; Iida, H.; Matsumoto, K.; Inoue, K.; Torii, K.;
Computer Software and Applications Conference, 2000. COMPSAC 2000. The 24th An
25-27 Oct. 2000 Page(s):191 - 197
AbstractPlus | Full Text: PDF(644 KB)   **IEEE CNF**

View Selected Items

Help    Contact Us    Privacy & S

Indexed by
**Inspec**

**IEEE Xplore** ® RELEASE 2.0

**Welcome United States Patent and Trademark Office**

**Search Results**                    BROWSE        SEARCH        IEEE XPLORE GUIDE

Results for "( incremental<in>ab ) <and> ( recompile<in>ab )"                    ✉ e-mail
Your search matched **2** of **1137806** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» **View Session History**

» **New Search**

» **Key**

**IEEE JNL** IEEE Journal or
           Magazine

**IEE JNL** IEE Journal or
          Magazine

**IEEE
CNF** IEEE Conference
     Proceeding

**IEE CNF** IEE Conference
           Proceeding

**IEEE
STD** IEEE Standard

**Modify Search**

( incremental<in>ab ) <and> ( recompile<in>ab )          [ » ]

☐  Check to search only within this results set

**Display Format:**  ◉ Citation   ○ Citation & Abstract

**Select      Article Information**

☐      1. **A macroscopic profile of program compilation and linking**
          Linton, M.A.; Quong, R.W.;
          Software Engineering, IEEE Transactions on
          Volume 15, Issue 4, April 1989 Page(s):427 - 436
          AbstractPlus | Full Text: PDF(520 KB)    **IEEE JNL**

☐      2. **Incremental netlist compilation for IKOS hardware logic simulator**
          Wang, K.; Chen, J.;
          ASIC Seminar and Exhibit, 1989. Proceedings., Second Annual IEEE
          25-28 Sept. 1989 Page(s):P9 - 3/1-4
          AbstractPlus | Full Text: PDF(260 KB)    **IEEE CNF**

[ View Selected Items ]

Help    Contact Us    Privacy & S

Indexed by
**Inspec** ®

**Welcome United States Patent and Trademark Office**

**Search Results**          BROWSE          SEARCH          IEEE XPLORE GUIDE

Results for "( compiler<in>ab ) <and> ( incremental<in>ab )"          ✉ e-mail
Your search matched **16** of **1137806** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search                    **Modify Search**

» **Key**                       ( compiler<in>ab ) <and> ( incremental<in>ab )          [»]

**IEEE JNL** IEEE Journal or      ☐ Check to search only within this results set
            Magazine
                                **Display Format:**   ◉ Citation   ○ Citation & Abstract
**IEE JNL**  IEE Journal or
            Magazine
**IEEE      IEEE Conference       **Select      Article Information**
CNF**       Proceeding
**IEE CNF** IEE Conference        ☐   1. **Incremental scanning and parsing with galaxy**
            Proceeding                  Beetem, J.F.; Beetem, A.F.;
**IEEE                                   Software Engineering, IEEE Transactions on
STD**       IEEE Standard                Volume 17, Issue 7, July 1991 Page(s):641 - 651

                                        AbstractPlus | Full Text: PDF(924 KB)   **IEEE JNL**

                                ☐   2. **The organization and performance of a TREAT-based production system compile**
                                        Miranker, D.P.; Lofaso, B.J.;
                                        Knowledge and Data Engineering, IEEE Transactions on
                                        Volume 3, Issue 1, March 1991 Page(s):3 - 10

                                        AbstractPlus | Full Text: PDF(712 KB)   **IEEE JNL**

                                ☐   3. **Segue: support for distributed graphical interfaces**
                                        Schaffner, S.C.; Borkan, M.;
                                        Computer
                                        Volume 21, Issue 12, Dec. 1988 Page(s):42 - 55

                                        AbstractPlus | Full Text: PDF(1076 KB)   **IEEE JNL**

                                ☐   4. **Incremental compilation for parallel logic verification systems**
                                        Tessier, R.; Jana, S.;
                                        Very Large Scale Integration (VLSI) Systems, IEEE Transactions on
                                        Volume 10, Issue 5, Oct. 2002 Page(s):623 - 636

                                        AbstractPlus | References | Full Text: PDF(1180 KB)   **IEEE JNL**

                                ☐   5. **Circuit compilers don't have to be slow**
                                        Diss, W.C.;
                                        Design Automation Conference, 1988. Proceedings., 25th ACM/IEEE
                                        12-15 June 1988 Page(s):622 - 627

                                        AbstractPlus | Full Text: PDF(296 KB)   **IEEE CNF**

                                ☐   6. **Fast incremental netlist compilation of hierarchical schematics**
                                        Jones, L.G.;
                                        Computer-Aided Design, 1989. ICCAD-89. Digest of Technical Papers., 1989 IEEE Inte
                                        Conference on
                                        5-9 Nov. 1989 Page(s):326 - 329

                                        AbstractPlus | Full Text: PDF(360 KB)   **IEEE CNF**

☐    **7. A strict monolithic array constructor**
Gao, G.R.; Yates, R.K.; Dennis, J.B.; Mullin, L.M.R.;
Parallel and Distributed Processing, 1990. Proceedings of the Second IEEE Symposium
9-13 Dec. 1990 Page(s):596 - 603

AbstractPlus | Full Text: PDF(600 KB)    **IEEE CNF**

☐    **8. Domain flow and streaming architectures**
Theodore, E.; Omtzigt, L.;
Application Specific Array Processors, 1990. Proceedings of the International Conferen
5-7 Sept. 1990 Page(s):438 - 447

AbstractPlus | Full Text: PDF(476 KB)    **IEEE CNF**

☐    **9. PDPS: a protocol design system by production systems approach**
Huang, C.-M.; Chang, Y.-I.; Liu, M.T.;
Systems Integration, 1990. Systems Integration '90., Proceedings of the First Internatio
on
23-26 April 1990 Page(s):186 - 195

AbstractPlus | Full Text: PDF(756 KB)    **IEEE CNF**

☐    **10. Hybrid incremental alias algorithms**
Marlowe, T.J.; Ryder, B.G.;
System Sciences, 1991. Proceedings of the Twenty-Fourth Annual Hawaii Internationa
Volume ii, 8-11 Jan. 1991 Page(s):428 - 437 vol.2

AbstractPlus | Full Text: PDF(984 KB)    **IEEE CNF**

☐    **11. The Synergetic Effect of Compiler, Architecture, and Manual Optimizations on the
of CFD on Multiprocessors**
Kuba, M.; Polychronopoulos, C.D.; Gallivan, K.;
Supercomputing, 1995. Proceedings of the IEEE/ACM SC95 Conference
03-06 Dec. 1995 Page(s):72 - 72

AbstractPlus | Full Text: PDF(168 KB)    **IEEE CNF**

☐    **12. Incremental compilation in the VCS environment**
Kripa Sundar, V.; Naik, A.V.; Chowdhury, D.R.;
Verilog HDL Conference and VHDL International Users Forum, 1998. IVC/VIUF. Proce
International
16-19 March 1998 Page(s):14 - 19

AbstractPlus | Full Text: PDF(24 KB)    **IEEE CNF**

☐    **13. Compiler/interpreter generator system LISA**
Mernik, M.; Lenic, M.; Avdicausevic, E.; Zumer, V.;
System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conferen
4-7 Jan. 2000 Page(s):10 pp.

AbstractPlus | Full Text: PDF(432 KB)    **IEEE CNF**

☐    **14. Developmental software environment that is applicable to small-size humanoids
humanoids**
Kanehiro, F.; Inaba, M.; Inoue, H.; Hirukawa, H.; Hirai, S.;
Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conferen
Volume 4, 2001 Page(s):4084 - 4089 vol.4

AbstractPlus | Full Text: PDF(612 KB)    **IEEE CNF**

☐    **15. Business compilers: towards supporting a highly re-configurable architectural st
oriented architecture**
Arsanjani, A.;
Software Maintenance, 2002. Proceedings. International Conference on
3-6 Oct. 2002 Page(s):287 - 288

AbstractPlus | Full Text: PDF(257 KB)    **IEEE CNF**

☐    **16. Grammatical approach to problem solving**
Henriques, P.R.; Kosar, T.; Mernik, M.; Pereira, M.J.V.; Zumer, V.;
Information Technology Interfaces, 2003. ITI 2003. Proceedings of the 25th Internation
16-19 June 2003 Page(s):645 - 650

AbstractPlus | Full Text: PDF(540 KB)    **IEEE CNF**

View Selected Items

Help    Contact Us    Privacy & S

Indexed by
**inspec**